数理解析研究所講究録
第 1894 巻 2014 年 41-44

2013 年度冬の LA シンポジウム [10]

グラフの経路固定サーバ割当問題に関する研究
The Server Supply-Assignment Problem on Graphs under a Given Routing Table

大日野 肇 * 伊藤 健洋 * 鈴木 顕 * 内澤 啓 † 周 暁 *
Hajime Oohino Takehiro Ito Akira Suzuki Kei Uchizawa Xiao Zhou

* 東北大学大学院情報科学研究科
Graduate School of Information Sciences, Tohoku University.
† 山形大学大学院理工学研究科
Graduate School of Science and Engineering, Yamagata University.

1 はじめに

昨今，コンピュータネットワークが高速・広帯域になるにつれ，高品質な動画や音声の配信など，多様でリッチなコンテンツの供給がなされるようになってきた．このようなコンテンツには十分に広い帯域の保証が必要であり，多数のユーザにサービスを安定供給するには全体として大きな帯域が必要となる．そこで近年は，同一コンテンツを配信するサーバを複数個所に配置することで，サーバの供給力が不足しないようにする工夫が行われている．しかし，回線の帯域には制限があり，あるユーザとサーバが帯域占有型の通信を行うと，その通信経路を共有する他のユーザが使用できる帯域が減少してしまう．したがって，ユーザの要求に対してコンテンツ配信を行うサーバをうまく割り当てることで，将来のユーザの要求にも応えられるようにしたい [4]．本稿では，このような状況をグラフにおける最適化問題として定式化し，研究を行った．

1.1 問題の定式化

点集合 V と辺集合 E からなるグラフを $G=(V,E)$ と書く．グラフ G の点集合と辺集合を，それぞれ $V(G)$ と $E(G)$ で記すこともある．$\mathbb{Z}_+$ を非負整数の集合とする．

グラフ G の点集合 V は，3 つの部分集合 V_S, V_U, V_R に分割される．点集合 V_S に含まれる各点は**サーバ点**と呼ばれ，点集合 V_U に含まれる各点は**ユーザ点**と呼ばれ，点集合 V_R に含まれる各点は**経由点**と呼ばれる．本稿では，各サーバ点は無限の供給能力を持つと仮定する．グラフ G の各辺 $e \in E$ には非負整数の**辺容量** $c(e) \in \mathbb{Z}_+$ が割り当てられているとする．ただし，以下では簡単のために，辺 $(u,v) \in E$ の辺容量 $c((u,v))$ を $c(u,v)$ と書く．

本稿では，サーバ点 $s \in V_S$ とユーザ点 $u \in V_U$ の任意の組について，G 上のパスが予め定められているとする．これを s から u への**通信経路**と呼ぶ．G の**通信経路表** $\mathcal{T}$ とは通信経路をまとめたものであり，任意のサーバ点 s から任意のユーザ点 u への通信経路は $\mathcal{T}(s,u)$ と表される．

今，あるユーザ点 $u \in V_U$ に対し，$n_s = |V_S|$ 個のサーバから通信を行うことを考えよう．通信経路表 $\mathcal{T}$

が予め与えられていることから，各サーバ点からの供給量を定めれば通信は定義できる．ユーザ点 $u \in V_U$ に対する**サーバ割当** $\boldsymbol{a} = (a_1, a_2, \ldots, a_{n_s}) \in \mathbb{Z}_+^{n_s}$ は，各サーバ点 $s_i \in V_S$ がユーザ点 u に通信経路 $\mathcal{T}(s_i, u)$ に沿って $a_i \in \mathbb{Z}_+$ だけ供給することを表す．ここで，$\mathbb{Z}_+^{n_s}$ は，非負整数の要素からなる n_s 次元ベクトルの集合を表す．なお，$a_i = 0$ となるサーバ s_i もあることに注意されたい．以降，$\boldsymbol{a}[i] = a_i$ と表記する．ユーザ点 $u \in V_U$ に対するサーバ割当 $\boldsymbol{a}$ が，全ての辺容量を超過しないとき，そのサーバ割当は**実行可能**であるという．すなわち，全ての $e \in E$ に対して

$$c(e) \geq \sum_{1 \leq i \leq n_s} \{\boldsymbol{a}[i] : e \in E(\mathcal{T}(s_i, u))\}$$

が成り立てばよい．

辺容量 c のグラフ G において，ユーザ点 $u \in V_U$ が n_s 個のサーバから受けられる最大の供給を，**ユーザ点 u の余力** $\mathsf{mar}(G, c, u) \in \mathbb{Z}_+$ と呼び，次のように定義する．

$$\mathsf{mar}(G, c, u) = \max\{ \sum_{1 \leq i \leq n_s} \boldsymbol{a}[i] : \boldsymbol{a} \text{ は } u \text{ に対する実行可能なサーバ割当}\}.$$

また，辺容量 c における**グラフ G の余力** $\mathsf{mar}(G, c)$ とは，ユーザ点 $u \in V_U$ の余力 $\mathsf{mar}(G, c, u)$ のうち最小のものと定義する．すなわち，

$$\mathsf{mar}(G, c) = \min_{u \in V_U} \mathsf{mar}(G, c, u)$$

である．

今，あるユーザ点 $r \in V_U$ が，合計で $d \in \mathbb{Z}_+$ の供給を受けたいという**要求** (r, d) をしたとしよう．このユーザ点 r を**要求点**と呼ぶ．$\sum_{1 \leq i \leq n_s} \boldsymbol{a}[i] = d$ なる r に対する実行可能なサーバ割当 $\boldsymbol{a} \in \mathbb{Z}_+^{n_s}$ において，**割当後辺容量** $c_{\boldsymbol{a}}(e)$ を各辺 $e \in E$ に対して次のように定義する．

$$c_{\boldsymbol{a}}(e) = c(e) - \sum_{1 \leq i \leq n_s} \{\boldsymbol{a}[i] : e \in E(\mathcal{T}(s_i, r))\}.$$

サーバ割当問題とは，グラフ G，辺容量 c および要求 (r, d) が与えられたとき，その要求 (r, d) を満たす実行可能なサーバ割当 $\boldsymbol{a}$ のうち，割当後辺容量 $c_{\boldsymbol{a}}(e)$ におけるグラフ G の余力を最大化するものを求める問題である．すなわち，要求 (r, d) を満たす最適なサーバ割当 $\boldsymbol{a}^*$ は，次式を満たす．

$$\mathsf{mar}(G, c_{\boldsymbol{a}^*}) = \max_{\boldsymbol{a}} \mathsf{mar}(G, c_{\boldsymbol{a}}).$$

上式では，要求 (r, d) を満たす全ての実行可能なサーバ割当 $\boldsymbol{a} \in \mathbb{Z}_+^{n_s}$ に対して最大値をとる．要求 (r, d) に対する最適解の値を $\mathsf{OPT}(r, d) = \mathsf{mar}(G, c_{\boldsymbol{a}^*})$ と表す．

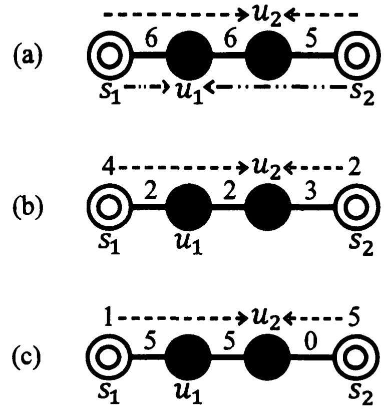

図 1: (a) サーバ割当問題の入力例，(b) 要求 $(u_2, 6)$ を満たすサーバ割当 $\boldsymbol{a}$，(c) 要求 $(u_2, 6)$ を満たす最適なサーバ割当 $\boldsymbol{a}^*$．

図 1(a) は，サーバ割当問題の入力グラフ G の例である．ここで，サーバ点は二重丸で，ユーザ点は影付きの丸で描かれている．なお，経由点はこの例にはない．また，辺容量は各辺の近くに与えられ，通信経路はそれぞれ点線で示されている．今，要求 $(u_2, 6)$ が与えられたとすると，図 1(b) と (c) はどちらもその要求を満たす実行可能なサーバ割当である．しかし，図 1(b) のサーバ割当 $\boldsymbol{a} = (4, 2)$ に対しては $\mathsf{mar}(G, c_{\boldsymbol{a}}) = \mathsf{mar}(G, c_{\boldsymbol{a}}, u_1) = 4$ であるが，図 1(c) のサーバ割当 $\boldsymbol{a}^* = (1, 5)$ に対しては $\mathsf{mar}(G, c_{\boldsymbol{a}^*}) = \mathsf{mar}(G, c_{\boldsymbol{a}^*}, u_1) = \mathsf{mar}(G, c_{\boldsymbol{a}^*}, u_2) = 5$ であり，これが最適解である．

サーバ割当問題は本稿で定義した問題であり，著者らが知る範囲においては，直接関連する結果はな

い．Kar ら [3] は，異なるモデル上で，本稿と似た観点に基づく研究をしている．そこでは，どの点もユーザ点とサーバ点を兼ねており，各ユーザ点はただ 1 つのサーバ点からしか供給を受けられず，通信経路も指定されていない．詳しくは，文献 [3] を参照されたい．

1.2 本稿の結果

本稿では，グラフクラスの観点から，サーバ割当問題の計算困難性と容易性を明らかにする．

まず，計算困難性として，カクタスに対してさえ，サーバ割当問題が NP 困難であることを示す．本稿の帰着は近似率を保持するように構成され，$\mathrm{P} \neq \mathrm{NP}$ の仮定の下では，どんな定数に対しても，カクタスには多項式時間の定数倍近似アルゴリズムが存在しないことも示せる．

次に，計算容易性として，木のサーバ割当問題を擬多項式時間で解くアルゴリズムを与える．アルゴリズムは，動的計画法に基づいており，$O(w_{\max}^4 n^3)$ 時間で最適解を求める．ここで，n は木の点数であり，$w_{\max}$ は木の辺容量の最大値である．したがって，$w_{\max}$ が n に関する多項式サイズであれば，本アルゴリズムは多項式時間で最適解を求める．

最後に，本稿では一般化されたサーバ割当問題も扱う．実際のコンピュータネットワークでは，1 つのユーザが通信できるサーバの個数には上限がある．そこで，サーバ割当問題において 1 点のユーザ点が通信できるサーバ点の個数を高々p 個に制限した問題を考える．本稿では，この供給サーバ数が制限された問題は，木に対してさえ NP 困難であることを示す．

2 カクタス

どの辺も高々1 つの閉路にしか含まれないようなグラフを**カクタス**という．本節では，カクタスのサーバ割当問題に対し，次の定理を与える．

定理 1. カクタスに対するサーバ割当問題は，NP 困難である．さらに，$\mathrm{P} \neq \mathrm{NP}$ の仮定の下では，どんな定数に対しても，多項式時間の定数倍近似アルゴリズムは存在しない．

定理 1 の証明として，MAXIMUM SET PACKING [2] と呼ばれる問題から，我々の問題への多項式時間帰着を与えた．MAXIMUM SET PACKING は NP 困難であることが知られており [1, 2]，また，$\mathrm{P} \neq \mathrm{NP}$ の仮定の下では，どんな定数に対しても，多項式時間の定数倍近似アルゴリズムは存在しないことが知られている [2]．本稿の帰着は，近似率を保持するものになっており，近似不可能性も証明できる．しかし，詳細は省略する．

3 木

本節では，次の定理を与える．

定理 2. 木 $T = (V, E)$ に対するサーバ割当問題は，$O(w_{\max}^4 n^3)$ 時間で解ける．ただし，$n = |V|$ であり，$w_{\max} = \max\{c(e) : e \in E\}$ である．

定理 2 の証明として，木のサーバ割当問題を $O(w_{\max}^4 n^3)$ 時間で解くアルゴリズムを与える．入力グラフが木であるとき，通信経路は一意に定まることに注意されたい．我々のアルゴリズムは動的計画法に基づいており，一般には指数通り考えられるサーバ割当 $\boldsymbol{a} = (a_1, a_2, \ldots, a_{n_s}) \in \mathbb{Z}_+^{n_s}$ のうち最適解に拡張しうる “候補解” のみを計算する．

$T = (V, E)$ を入力木とし，c を T の辺容量とする．与えられた要求 (r, d) に対し，T を要求点 r を根とする根付き木とみなす．このとき，各点 $v \in V$ に対して，v を根とし，v とその子孫全てを含む部分木を $T_v = (V_v, E_v)$ と書く．T_v 内部に含まれるサーバ点の割当 $\boldsymbol{x} = (x_1, x_2, \ldots, x_{|V_v \cap V_S|}) \in \mathbb{Z}_+^{|V_v \cap V_S|}$ と非負整数 $y \in \mathbb{Z}_+$ に対して，次のような（擬似的な）部分木 $T_v(\boldsymbol{x}, y)$ を定義する．**$(\boldsymbol{x}, y)$-部分木** $T_v(\boldsymbol{x}, y)$ とは，部分木 T_v にダミーのサーバ点 s' を加え，容量 y の辺で s' と r を接続したものである．すなわち，$V(T_v(\boldsymbol{x}, y)) = V_v \cup \{s'\}$ であり，$E(T_v(\boldsymbol{x}, y)) =$

$E_v \cup \{(s', r)\}$ である．$T_v(\boldsymbol{x}, y)$ の辺容量 c' は，各辺 $(u, v) \in E(T_v(\boldsymbol{x}, y))$ に対して，次のように定義される．

$$c'(u,v) = \begin{cases} y & (u,v) = (s',r) \text{ のとき,} \\ c_{\boldsymbol{x}}(u,v) & \text{その他.} \end{cases}$$

ここで，T_v 内部に含まれる辺に対する辺容量 c' が $\boldsymbol{x}$ に依存していることに注意されたい．したがって，$(\boldsymbol{x}, y)$-部分木 $T_v(\boldsymbol{x}, y)$ は，部分木 T_v 内部にある $|V_v \cap V_S|$ 個のサーバ点から $\boldsymbol{x}$ の割当をし，さらに T_v 外部から y だけ供給が受けられる状況を表している．すなわち，木 T の要求点 r に対して，$x_1+x_2+\cdots+x_{|V_v \cap V_S|}$ の供給を部分木 T_v から行う．

本稿のアルゴリズムのアイディアは，部分木 $T_v = (V_v, E_v)$ と非負整数 $x, y \in \mathbb{Z}_+$ が指定されたとき，$x_1 + x_2 + \cdots + x_{|V_v \cap V_S|} = x$ となるような T_v 内部のサーバの割当 $\boldsymbol{x} = (x_1, x_2, \ldots, x_{|V_v \cap V_S|}) \in \mathbb{Z}_+^{|V_v \cap V_S|}$ のうち，$(\boldsymbol{x}, y)$-部分木 $T_v(\boldsymbol{x}, y)$ の余力を最大化するものを 1 つだけ候補解として計算することである．すなわち，各部分木 $T_v = (V_v, E_v)$ と全ての非負整数 $x, y \in \mathbb{Z}_+$ に対し，

$$\begin{aligned} f(T_v, x, y) \;\; = \;\; & \max_{\boldsymbol{x}} \{ \mathsf{mar}(T_v(\boldsymbol{x}, y), c') : \\ & x = x_1 + x_2 + \cdots + x_{|V_v \cap V_S|} \} \end{aligned}$$

と定義される値 $f(T_v, x, y)$ を計算する．本稿のアルゴリズムは，動的計画法に基づき，木の葉から根 r に向かって，順に $f(T_v, x, y)$ を計算する．最終的に，要求 (r, d) に対する最適解 $\mathsf{OPT}(r, d)$ は，$T = T_r$ であることに注意すれば，

$$\mathsf{OPT}(r, d) = f(T_r, d, 0)$$

と計算することができる．本稿では，具体的な $f(T_v, x, y)$ の計算の仕方は省略する．

木 T の各点 $v \in V$ において，アルゴリズムは全ての非負整数 $x, y \in \mathbb{Z}_+$ に対し $f(T_v, x, y)$ の値を計算する．しかし実際には，x については $0 \le x \le d$ の範囲に対して，y については $0 \le y \le w_{\max}$ の範囲に対して計算すれば十分である．したがって，各点 v に対し $O(dw_{\max})$ 個の値を計算することになる．部分木で求めた候補解を基に，より大きな部分木の候補解を求めることは，$O(d^2 w_{\max}^2)$ 時間でできる．このような更新は $n-1$ 回で済むことが示せるため，根 r に対して $f(T_r, d, 0)$ の値は $O(d^2 w_{\max}^2 n)$ 時間で求めることができる．$d \le w_{\max} n$ であるから，アルゴリズム全体の計算時間は $O(w_{\max}^4 n^3)$ 時間となる．

4 供給サーバ数が制限された問題

本節では，正整数 p が与えられ，各ユーザ点が供給を受けられるサーバ点の個数が高々 p 個に制限されたサーバ割当問題を扱う．このような問題を**供給サーバ数制限付サーバ割当問題**と呼ぶ．本節では，次の定理を与える．

定理 3. 供給サーバ数制限付サーバ割当問題は，木に対して NP 困難である．

定理 3 の証明として，NP 完全であることが知られている PARTITION 問題 [1] から，我々の問題への多項式時間帰着を与えた．しかし，詳細は省略する．

参考文献

[1] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.

[2] J. Håstad, Clique is hard to approximate within $n^{1-\varepsilon}$, Acta Math. 182, pp. 105–142, 1999.

[3] K. Kar, M. Kodialam and T. V. Lakshman, Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications, IEEE J. Selected Areas in Communications 18, pp. 2566–2579, 2000.

[4] 古谷 快, 西山 大樹, 加藤 寧, 野村 啓仁, 矢田 健, 山田 博司, コンテンツ配信ネットワークにおけるトラヒック間作用を考慮したサーバ選択法に関する一検討, 電子情報通信学会総合大会講演論文集，vol. 2011 年通信，no. 2，p. 35，2011.